

In the Claims

Please amend claims 1, 8 and 15 as follows:

1. (currently amended) A method in a computer system for assessing the relative complexity of different options for performing a task by the computer system, the method comprising the steps of:

storing programming instructions on a storage medium of the computer system;

executing the instructions by the computer system, wherein the executing causes the computer system to implement a method comprising the steps of:

defining the task as a sequenced set data structure that specifies actions of the task, and sequence information that specifies the order in which particular actions are to be performed;

storing recipes available for performing constituent actions of the task as sequenced set data structures that specify subactions of the recipes for the constituent actions, and sequence information that specifies the order in which the subactions are to be performed;

determining complexity measures associated with performing the task using different combinations of recipes for constituent actions of the task, based upon complexity measures of actions specified by respective combinations of available recipes;

delegating the defined task to a primary agent for execution of the task by at least one of the primary agent and one or more contracting agents, wherein the complexity of the action is characterized by mutually exclusive measures comprising (i) the presence of recipes with the primary agent for performing the action and also the presence of agents for contracting out the action; (ii) the presence of recipes with the primary agent and the absence of agents for

contracting; (iii) the presence of agents for contracting and the absence of recipes with the primary agent; and (iv) the absence of both agents and recipes; and

presenting a report of the complexity measures associated with performing the task.

2. (original) The method as claimed in claim 1, wherein complexity measures for actions are defined in terms of the complexity measures of available recipes for performing the actions, and complexity measures for recipes are defined in terms of the complexity of the subactions of the recipe.

3. (original) The method as claimed in claim 1, further comprising the steps of:

determining predetermined complexity measures for basic actions that are not specified by a recipe; and

determining specified complexity measures for contracted actions that are performed by a different agent.

4. (original) The method as claimed in claim 1, further comprising the alternating steps of:

updating complexity measures for recipes in relation to actions whose complexity measures are determined; and

updating complexity measures for actions in relation to recipes whose complexity measures are updated.

5. (original) The method as claimed in claim 1, wherein the complexity measures associated with a particular action performed by a particular agent is based upon the complexity measures for each of the recipes for that action.

6. (previously presented) The method as claimed in claim 1, wherein the sequence information that specifies the order in which particular actions are to be performed specifies, for pairs of actions, that one specified action is sequenced before another specified action.

7. (original) The method as claimed in claim 1, further comprising the step of defining a sequenced set data structure as  $S = (A, M)$ , in which  $A$  is a multi-set element and  $M$  is a sequencing relation that specifies an ordered sequence of the elements  $A$  in the sequenced set  $S$ .

8. (currently amended) The method as claimed in claim 7, further comprising the step of defining a sequencing relation for the sequenced set data structure  $S$  for two elements  $a_i$  and  $a_j$  of multi-set element  $A$ , such that  $a_i$  is sequenced before  $a_j$  in set  $A$  under the relation  $M$ .

9. (cancelled)

10. (previously presented) The method as claimed in claim 1, wherein a plurality of series of actions exist for performing the defined task, each of the series of actions having a corresponding complexity, and the method further comprises the step of:

performing the defined task by executing a selected one of the series of actions, wherein the complexity of the selected series of tasks is less than the complexities of the other series of tasks of the plurality of series of tasks.

11-14. (canceled)

15. (currently amended) A method in a computer system for assessing the relative complexity of different options for performing a task-by the computer system, the method comprising the steps of:

storing programming instructions on a storage medium of the computer system;  
executing the instructions by the computer system, wherein the executing causes the computer system to implement a method comprising the steps of:

defining the task as a sequenced set data structure that specifies actions of the task, and sequence information that specifies the order in which particular actions are to be performed;

storing recipes available for performing constituent actions of the task as sequenced set data structures that specify subactions of the recipes for the constituent actions, and sequence information that specifies the order in which the subactions are to be performed;

delegating the defined task to a primary agent for execution of the task by at least one of the primary agent and one or more contracting agents, wherein the complexity of the action is characterized by mutually exclusive measures comprising (i) the presence of recipes with the primary agent for performing the action and also the presence of agents for contracting out the action; (ii) the presence of recipes with the primary agent and the absence of agents for contracting; (iii) the presence of agents for contracting and the absence of recipes with the primary agent; and (iv) the absence of both agents and recipes; and

determining complexity measures associated with performing the task using different combinations of recipes for constituent actions of the task, based upon complexity measures of actions specified by respective combinations of available recipes;

wherein a plurality of series of actions exist for performing the defined task, each of the series of actions having a corresponding complexity, and the method further comprises the step of:

performing the defined task by executing a selected one of the series of actions, wherein the complexity of the selected series of tasks is less than the complexities of the other series of tasks of the plurality of series of tasks, and wherein the defined task includes a task for administering a networked computer system:

16. (previously presented) The method as claimed in claim 15, wherein complexity measures for actions are defined in terms of the complexity measures of available recipes for performing the actions, and complexity measures for recipes are defined in terms of the complexity of the subactions of the recipe.

17. (previously presented) The method as claimed in claim 15, further comprising the steps of:

determining predetermined complexity measures for basic actions that are not specified by a recipe; and

determining specified complexity measures for contracted actions that are performed by a different agent.

18. (previously presented) The method as claimed in claim 15, further comprising the alternating steps of:

updating complexity measures for recipes in relation to actions whose complexity measures are determined; and

updating complexity measures for actions in relation to recipes whose complexity measures are updated.

19. (previously presented) The method as claimed in claim 15, wherein the complexity measures associated with a particular action performed by a particular agent is based upon the complexity measures for each of the recipes for that action.

20. (previously presented) The method as claimed in claim 15, wherein the sequence information that specifies the order in which particular actions are to be performed specifies, for pairs of actions, that one specified action is sequenced before another specified action.

21. (previously presented) The method as claimed in claim 15, further comprising the step of defining a sequenced set data structure as  $S = (A, M)$ , in which  $A$  is a multi-set element and  $M$  is a sequencing relation that specifies an ordered sequence of the elements  $A$  in the sequenced set  $S$ .

22. (previously presented) The method as claim in claim 21, further comprising the step of defining a sequencing relation for the sequenced set data structure  $S$  for two elements  $a_i$  and  $a_j$  of multi-set element  $A$ , such that  $a_i$  is sequenced before  $a_j$  in set  $A$  under the relation  $M$ .

23. (cancelled)